

EPROM CARTRIDGE DOWNLOAD ROUTINE

This routine will take a basic program stored in the 16K memory block from C000-FFFFh and move it down into the normal Basic ram area. (4009h and on) More than one program may be stored on a board in this block. POKE 16417 the number of the program desired to be downloaded.

The first programs' number is zero, the seconds' is one, etc. so that the routine will default on power up to download the first program in the block. (Because 16417=00 on power up) The programs should be stored on eprom with no separating bytes. If the desired program, other than program zero, does not exist on the cartridge, the computer will return with report "E", program requested is not on the cartridge board. If there are no programs available, because the cartridge is not inserted into the expansion board, the cartridge board enable switch is off, or the memory board's 48/64K switch is on, the computer will crash.

2172 CDE702	STRT CALL TFAS	;Enter "TEMP FAST" mode
2175 2100C0	LD HL,C000	;Point 'HL' at cartridge slot
2178 110B00	AGIN LD DE,000B	;Set 'DE' for displcmnt to ELINE
217B 19	ADD HL,DE	;Point 'HL' at ELINE
217C 5E	EDPA LD E,(HL)	; 'DE'=ELINE
217D 23	INC HL	; address
217E 56	LD D,(HL)	;
217F 3A2140	LD A,(4021)	;Get program no. to download
2182 A7	AND A	; 'A'=00?
2183 EB	EX DE,HL	; 'HL'=add ELINE/'DE'=ABS ELINE
2184 2019	JR NZ NEXT	;Jump if wrong program
2186 010940	LD BC,VERSN	;Set 'BC' for calculation
2189 C5	PUSH BC	;Save 4009h on the stack
218A ED42	SBC HL,BC	; 'HL'=byte count
218C 44	LD B,H	; 'BC'=byte
218D 4D	LD C,L	; count
218E EB	EX DE,HL	; 'HL'=ABS address ELINE+1
218F 110C00	LD DE,000C	;Set 'DE' for calculation
2192 ED52	SBC HL,DE	; 'HL'=start of the program
2194 D1	DEST POP DE	; 'DE'=dest (4009h)
2195 EDB0	LDIR	;Move the program
2197 E1	POP HL	;Discard previous return address
2198 217606	LD HL,0676	;Set HL for new RETURN address
219B E5	PUSH HL	;Put new return address on stack
219C C30702	JP SLO?	;Return to 0676h via SLO?
219F 011540	NEXT LD BC,4015	;Set BC for calculation
21A2 ED42	SBC HL,BC	;HL=no. of bytes to skip
21A4 19	ADD HL,DE	;HL=start of next program
21A5 FD3521	DEC (IY+21)	;Decrement program counter
21A8 7E	LD A,(HL)	;Get first byte of next program
21A9 FEFF	CP FF	;Is another program there?
21AB 20CB	JR NZ AGIN	;Jump if there is
21AD CF	RST 08H	; Return w/report "E"
21AE 0D	REPORT E	; Program no. not available

BASIC PROGRAM PROGRAMMER ROUTINE

This routine will program a 2764 eeprom with the current program in memory. The amount of data "SAVED" to eeprom will be the same amount as if the normal SAVE command was used, so if you don't need the variables in a program, CLEAR before calling to save eeprom space. The USR routine may be called from within a program, if desired, to make the program self running.

POKE 16507 and 16508 either: The location in the eeprom you wish the routine to start programming at. (For more than one program on an eeprom--default starting address is 2000h) or The location in ram you wish the routine to start from. (In case the program is greater than 8K in length and this is the second eeprom to be programmed.) If the last eeprom was entirely loaded with the beginning of the current program in memory, then POKE 16507,9 and POKE 16508,96--making the first address to be programmed from this time 6009h. Default source address used by this routine is 4009h. (VERSN) POKE 16507 with the lo byte of the 16 bit address and 16508 with the hi byte. (The same method used when changing ramtop, lo byte first + hi byte last) Be sure both 16507 + 16508 are zero if you are using the default addresses!!!

```

21AF CDE702      STRT CALL 02E7      ;Enter "TEMP FAST" mode
21B2 2A7B40      LD HL,(407B)      ;Get user pass bytes
21B5 AF          XOR A              ;Let 'A'=00
21B6 327B40      LD (407B),A        ;Clear user
21B9 327C40      LD (407C),A        ;pass bytes
21BC 7C          LD A,H             ;'A'=hi pass byte
21BD FE20        DFT? CP 20         ;Passed address >1FFF?
21BF 3005        JR NC SRC?         ;Jump for further tests if true
21C1 210940      LD HL,VERSN        ;'HL'=default src
21C4 180A        JR DEST            ;Continue at 21D0
21C6 FE40        SRC? CP 40         ;Address passed >3FFF?
21C8 3006        JR NC DEST         ;Jump if true. Addr. is for src
21CA EB          EX DE,HL           ;'DE'=passed dest
21CB 210940      LD HL,VERSN        ;'HL'=default src
21CE 1803        JR CONT            ;Continue at 21D3
21D0 110020      DEST LD DE,2000    ;'DE'=default dest
21D3 EB          CONT EX DE,HL       ;'DE'=src 'HL'=dest
21D4 E5          PUSH HL            ;Save dest
21D5 2A1440      LD HL,(ELINE)      ;Get last byte to move-1
21D8 A7          AND A              ;Clear carry
21D9 ED52        SBC HL,DE           ;'HL'=byte count
21DB 44          LD B,H              ;'BC'=byte
21DC 4D          LD C,L              ;count
21DD EB          EX DE,HL           ;'HL'=src
21DE D1          POP DE              ;Restore dest
21DF EDA0        MBYT LDI           ;Program a byte
21E1 E20702      JP PD SLO?         ;Return via "SLO?" if done
21E4 E5          PUSH HL            ;Save src
21E5 2100C0      LD HL,C000         ;Set 'HL' for test
21E8 19          ADD HL,DE           ;Dest>3FFF?
21E9 3002        JR NC NOER         ;Continue at 21ED if not
21EB CF          RST 08H            ;Return with report "G"
21EC 0F          REPORT G           ;Dest>3FFF
21ED D5          NOER PUSH DE        ;Save dest
21EE C5          PUSH BC            ;Save byte count
21EF 210400      LD HL,0004         ;Set 'HL' for the pause
21F2 CD2D02      CALL 022D          ;PAUSE 4
21F5 FD3635FF    LD (IY+35),FF      ;Set FRAMS-hi
21F9 CD4B0F      CALL 0F4B          ;Set DBNC
21FC C1          POP BC             ;Restore byte count
21FD D1          POP DE             ;Restore dest
21FE E1          POP HL             ;Restore src
21FF 18DE        JR MBYT            ;Loop to send another byte

```